IN THE UNITED STATES PATENT AND TRADEMARK OFFICE

Applicant	:	Matthias Wandel; Gary P. Mousseau
Title	:	System and Method for Implementing Local Base Stations
Application No.	:	09/891,038
Filing Date	:	6/25/01
Examiner	:	Greg C. Bengzon
Group Art Unit	:	2144
Attorney Docket	:	555255012248

Box AF
Commissioner for Patents
P.O. Box 1450
Alexandria, VA 22313-1450

DECLARATION UNDER 37 CFR §1.131

We, Matthias Wandel and Gary P. Mousseau, are the named inventors of the subject application.

We conceived of the subject matter of claims 67-70 prior to the June 26, 2000 filing date of U.S. Patent No. 6,804,221 to Magret et al, as evidenced by Exhibits A and B. Exhibit A is an invention disclosure of matter that was conceived by us and that fully supports claims 67-70. Exhibit B is a cover letter from an employee of the assignee, Research in Motion, forwarding the invention disclosure to the law firm of Jones Day. The cover letter is dated prior to June 26, 2000. This confirms that the invention disclosure, and thus our conception of claims 67-70, predates June 26, 2000.

We hereby declare that all statements made herein of our own knowledge are true and that all statements made on information and belief are believed to be true; and further that these statements were made with the knowledge that willful false statements and the like so made are punishable by fine or imprisonment, or both, under Section 1001 of Title 18 of the United States Code, and that such willful false statements may jeopardize the validity of the application or any patent issued thereon.

Matthias Wandel	Sept 28 2006	Gary P. Mousseau	29. Sept. 06
Matthias Wandel	date	Gary P. Mousseau	date

EXHIBIT A (22 pages)

Title: **SYSTEM AND METHOD FOR IMPLEMENTING BASE STATIONS USING PERSONAL COMPUTERS**

Inventor(s): **WANDEL, Matthias**

Assignee: **Research In Motion Limited**

BACKGROUND OF THE INVENTION

Field of the Invention

This invention relates to base station implementation in a personal or networked computer.

Uses of the Present Invention

1. End of line test base station

There is a need for a device that allows end customers to end of line test mobile communication devices such a Research In Motion Handheld or OEM radio modems integrated into a OEM product. Such a device could be in the form of a test base station that could be ship to end customers by a communication device manufacturer. There is a need to provide a device that is flexible, and less based on antiquated and custom hardware.

1.1. Customer site base station

Mobile communication manufacturers are seeing congestion at large customer sites, where a large number of *RIM* Wireless Handheld Devices ("mobile devices") are used in a small area. Creating base stations that could connect to their BlackBerry™ server, and making special provisions to allow modems to roam to such a base station when used with BlackBerry would allow deployment of additional customer owned base stations to alleviate the problem.

1.2. A platform to evolve Mobitex on

There are many ideas in terms of modifications to the ROSI protocols, and signal processing that RIM is interested in exploring. Having our own base station hardware would allow RIM to evolve the ROSI protocol.

BRIEF DESCRIPTION OF THE DRAWINGS

In order that the invention may be more clearly understood, preferred embodiments thereof will now be described in detail by way of example, with reference to the accompanying drawings, in which:

Fig. 1 is an overall system diagram of the present invention;
Fig. 2 illustrates a network communications topology;
Fig. 3 illustrates Base station software architecture;
Fig. 4 illustrates a regular MPAK delivery;

Fig. 5 illustrates how a mobile sends to a mobile route not on base and not in route cache;

Fig. 6 illustrates how a mobile roams to a new base in absence of traffic;

Fig. 7 illustrates how a mobile sends to a mobile that has just moved; and,

Fig. 8 illustrates an A-node MPAK state machine.

DETAILED DESCRIPTION OF THE PREFERRED EMBODIMENT

2. Technical approach

2.1. PC hardware based

When Ericsson designed Mobitex in the 1980's, all components were designed for Mobitex, including the networking infrastructure and RF transceivers. Much of their current problems evolve around the networking aspect of the modem, and keeping software for antiquated platforms up to date.

The present invention will now be described in conjunction with Figures incorporated herein. Figure 1 is an overall system diagram of the present invention. By using ordinary PCs **10**, with a third party OS, the whole digital hardware, operating system, and networking protocol issues are already present. Advantageously, commercially available, off-the-shelf components are combined and integrated to produce a base station **6** that allows two-way communication with a plurality of mobile devices **8**. As illustrated in Figure 1, the base station is built from, preferably, a general-purpose personal computer or networked computer **10** (collectively "PC"). Using a popular PC platform also ensures that the latest in terms good quality tools will be available for our development. Given the power of modern PC's even the signal processing can easily be done without additional hardware, or even writing assembly language code. If additional performance is needed, MMX assembly language may be incorporated.

The PC **10** includes a full duplex sound card **12** operable therewith. Configured to the sound card is a transmitter **22** and a receiver **20**. The transmitter and receiver facilitate the radio communication between the base station and the mobile devices.

The PC is then programmed with specific base station software modules such as, but not limited to, the following software modules: transmit task **24**, receive task **26**, link layer task **28**, network task **30**, and UDP communication task **32**.

It is to be understood, there may be a plurality of the base stations in a particular locale (such as at large corporate building). The present invention allows for mobile devices to roam seamlessly between one base station and another.

2.2. PC Use sound cards for receive and transmit

A full duplex sound card **12** can provide all the analog I/O we need for producing a baseband modulated stream **14**, and demodulating an incoming baseband signal **16**.

However, care must be taken in selecting an appropriate sound card model for the following

reasons:

- Not all sound card have a sufficiently low output frequency cutoff for to transmit GMSK cleanly
- Not all sound card have a sufficiently low input frequency cutoff for to transmit GMSK cleanly
- Sound cards are not required to be frequency accurate to 10 ppm as is desirable for Mobitex. Frequency accuracy needs to be considered and possibly calibrated for in the software modules.

The current experimental RIM base stations use an older style Soundblaster™ PCI128 sound card. This card has been found satisfactory in every respect.

2.3. Linux OS based

A number of operating systems could potentially be run on a PC platform for this purpose. However, Linux's open source is the most appealing, especially if OS customizations are needed. Note that it is not a requirement that the OS be strictly 'real time'.

UNIX device 'files' & getting up and running

What is needed from the OS and hardware first and foremost is to receive and transmit a continuous sample stream. The UNIX model of mapping devices as IO files would allow quick implementation without having to modify or hack with sound card programming or Direct X SDKs. As we add custom hardware, we may later find that we have to provide our own mechanism for getting samples in and out.

Real time responsiveness and Linux

Linux is not a real time operating system. Fortunately, the base station (unlike a mobile) does not have very hard real time constraints. Knowing that the OS will respond quickly should be sufficient for our purposes, as lack of OS responsiveness will only reduce efficiency, but not cause system failure. Having more than enough CPU horsepower (as is the case with PC hardware) generally makes the system very responsive.

2.4. Eventually build our own RF for it

Although much of the software development would be based on standard PC hardware and purchased transceiver hardware (such as an HP 8920 RF test set), it is preferable that a customized RF transceiver is implemented. Using customized RF hardware would allow for lower cost per unit, as well as implementing IQ receiving and intricate signal processing.

3. Implementation phases

3.1. Phase 1: Proof of concept / End of line test

Advantageously, the proof of concept has been based entirely on off-the-shelf hardware.

Based on ordinary PC hardware.

Although the goal is to eventually use an embedded PC, for development, and end of line testing, having a screen, keyboard, and disk drives available would be very desirable. As such, until the concept is proven and sufficient momentum acquired, there is little point in

moving to an embedded PC board.

Use sound cards to do analog sample streams in and out

A sound card is the choice for sample streams to and from the computer. Sound cards are already well supported, and can readily be purchased.

Use existing hardware for RF transceiver

Either use an 8920 RF test set as the RF transceiver, or an AOR 3000 as the receiver **20** and an HP signal generator as the transmitter **22.** It should be possible to connect these directly to the sound card analog inputs and outputs. Connecting via HPIB to the 8920 RF test set would also be useful for setting up unusual RF circumstances for end of line RF testing.

No fancy signal processing

If sound cards and 8920 RF test set are used for the receiver, there is little point in applying intricate signal processing techniques, as the system is already very sub-optimal for sensitivity. Most fancy signal-processing techniques rely on having an IQ signal available. As such, exploring signal processing algorithms would have to wait until we build our own IQ receiver.

3.2. Phase 2:Networking evolution

One of the bigger problems of building a network is the whole wide area networking aspects. Although IP protocols can handle wide area networking, they do not natively handle roaming of the sort that Mobitex provides. As such, some software needs to be written on top of TCP/IP to deal with rapidly moving mobile devices.

Connecting base stations together & roaming handling

One of the interesting stages of development will be to build a test network consisting of several base stations and servers, and do testing of roaming, activation's, traffic, etc.

Integrating with BlackBerry™ for customer site base station

For a customer site base station, the base station, or sub-network must be connected in with the Blackberry server, and provisions made in the BlackBerry server, as well as handhelds, to deal with mobiles moving between standard Mobitex, and the local Mobitex network

Phase 3: A full Mobitex-like WAN

Once bases are working together reliably, and communicating with Blackberry, the next step in the evolution is to design a whole WAN based on the present invention technology.

Subscription server

This will involve writing a scalable subscription server, and various other tools required to administer a large network. The subscription server keeps track of where everyone is currently located and what their state is, much like a HLR / VLR in a GSM network.

Better administration tools

A WAN would also include such things as a good wireless subscription management scheme.

3.3. Hardware evolution

Initial hardware would be based entirely on purchased components. This would be acceptable for an end of line testing solution. However, hardware components can be engineered to make it a viable commercial product for uses other than end of line testing.

Replace the sound cards

Sound cards are not built for use in a base station, and once the device becomes a product, the sound card might have to be improved on.

Migrate to an embedded PC platform

Once we are ready to have our platform used for infrastructure purposes, migrating onto a more reliable hardware PC platform becomes important. High quality embedded PCs will probably provide the hardware reliability we need. This could also involve moving to a flash based file store, instead of using a hard drive.

Build IQ receiver & use MMX to do fancy processing

Our own transceiver will be necessary for this to be a viable product. Once we use our own RF transceiver, we can receive in IQ, and then we can start playing fancy games with signal processing on the receive side. A PC is particularly suitable for some of the tricks in terms of combining successive received signals, as a PC, unlike a dedicated DSP, has many megabytes of memory to store undecodable signal fragments in.

4. WAN topology considerations

Preferably, everything will connect together with an IP network. However, nothing in the TCP/IP protocols make allowances for subscriptions roaming quickly and randomly through a large WAN. As such, some roaming / routing layer on top of TCP/IP will be required to handle roaming issues.

Because geographical issues are hidden when using an Internet network as the fundamental communications mechanism, it may not be necessary to build a multi-layered tree shaped network. Instead, for the sake of simplicity, having only two layers in the network would be more appropriate. One layer being the base stations, the other being the servers containing subscription and location information.

4.1. Some general principles

Use IP protocols where practical

Ideally, each base connects to the Mobitex network by making connections using IP over an Internet, preferably an existing private network. Thus, the overhead of maintaining landlines and WAN hardware is maintained by the client implementing the base station. Also, the overhead of IP protocols such as TCP/IP is acceptable when tunneling MPAKS through the Internet, as any MPAK traffic is only a trickle compared to other Internet traffic, and internet capacity. Reliance on IP protocols also simplifies routing and scalability, as these problems has already been addressed in Internet technology.

No backup for base stations

Although redundancy increases reliability, redundancy for base stations would be very expensive, as the base stations will be the most numerous element in the entire network. When base stations provide areas of overlapping coverage, as they do in most urban centers, the failure of a base station can already be accommodated for by mobiles roaming to a different channel. Thus, in terms of network topology, redundancy and replication should be limited to the server side.

Third party servers do not connect directly to base stations

Preferably, third party servers should not be allowed to connect to the IP base stations. This is because mobiles will roam to different bases. If a third party server connects to a base, and mobiles roam to different bases, the base would be unnecessarily loaded, and an unnecessary point of failure. Expecting third party servers to connect to base stations where mobiles have roamed to is also not a good idea, as this would create problems in terms of revising base-server protocols, and as well as problems in terms of base stations failing and finding forwarding IP addresses.

Base stations initiate internal network connections

Preferably, whenever a base is connected or brought up on the network, it is the base that contacts the servers, not vice versa. Any routing tables that servers may contain will be built up through base stations coming up and contacting the servers. Thus, if a base fails temporarily, it is able to re-initiate contact as soon as it comes back up. Also, because bases are more numerous, one is much more likely to

Master Subscription info always in the subscription server

The master copy of all mobile information is kept in the network. Thus, when it is found that a mobile has roamed to a new location, it will not be necessary to request the information from the last base station. Another advantage is that if a base station rests, or is unable to push a subscription up when its full, the subscription can be pushed back down again by the base. This requires that the network's copy be updated whenever a base finds that any of the subscription's status has changed.

4.2. Network communications topology

Figure 2 illustrates a network communications topology. Many base stations, and one or more subscription servers connect to the Internet. FSTs connect through an FST base to the rest of the network.

Base stations can communicate peer to peer. All MPAKs are routed peer to peer, with no MPAKs going through the subscription servers. The FST base handles semantics of peer-to-peer communications between base stations on the FST's behalf.

The subscription server maintains an up to date master copy of the status of any mobile device, and is able to give any mobile the most recent information as to which base station a mobile is currently roamed to. The server also maintains items such as currently valid MAN numbers and how their usage may affect billing.

Thus, *ALL MPAKS are routed peer to peer between base stations and all status information goes through the subscription server.*

5. Base station software architecture

Each base keeps two tables:

Subscription list
This is the list of all mobiles currently active (or last known to be active) on that base. The list is used to verify that mobiles are valid mobiles every time a mobile communicates, without querying the server. Entries are added to this list as mobiles roam on to the base, and deleted from the list as the server notifies the base that the mobile has appeared elsewhere.

Route Cache
Every time a base needs to send to another subscription, it checks its route cache to see if it already knows the IP address of the base where that mobile was seen last. If an entry exists, the MPAK is sent directly to the next base. If no entry exits, the base must obtain the route information from the server first. If the route used is obsolete, the base sent to will notify the sender that the route is obsolete, and the sender obtains a new route from the server.

Figure 3 illustrates base station software architecture.

6. Base station-network signals

Because of the large number of possible paths between the base stations and servers, permanently open TCP connections would overload the system, and temporary TCP connections would be very inefficient. As such, UDP is probably the best way to go. However, when using UDP, acknowledgements must be handled by the application.

In this scheme, some UDP packets have acknowledgements indicating that they have been received, while other UDP packets have acknowledgements containing requested data. Generally, if data is requested, the acknowledgement and the result of the query are in fact the same UDP datagram.

Infoless ACK indicates that an ack must be sent in response to the message, but the ACK can be a generic ACK of sorts, as it carries no type specific data.

6.1. Messages exchanged between bases and subscription servers

Mobile info request
After a base reports that a mobile has arrived, the network will send the subscription data for that modem. This may include information such as that the subscription is invalid, or that the mobile should be ignored. Includes order to send a DIE packet to a modem.

Mobile info
Subscription data in response to Mobile info request. Also spontaneously generated by the network to push a subscription back into a base, or change the state of a subscription, such as indicating that the mobile now has mail waiting for it.

Remove subscription
The server is instructing a base that one of its subscriptions has moved elsewhere. (infoless ACK)

Unload Subscription
Base is overflowing its list of active subscription, and needs to push subscriptions that have not been used for a long time back into the server. (infoless ACK)

Mobile route request
A Base needs to determine the IP address of the base where the destination MAN is believed to be at. Server looks up IP address of base where destination MAN is believed to be.

Mobile route reply
Server responds with IP address of base where destination MAN number is believed to be. (no ACK)

Mobile Inactive
Mobile has become inactive. Do not route data here. (infoless ACK)

Mobile Active
An inactive mobile has become active. (infoless ACK)

Mobile mail pending
Base sends to server indicating it has mail for a specific mobile. Server then notifies base where mobile resides that the mailbox flag should be set, and who should be notified when mobile shows up by sending it a new mobile info reply message. (infoless ACK)

Base offline
The base intentionally goes offline (for service purposes). Server flags all mobiles on a base as 'unknown location', base goes off line long enough for all modems to loose coverage.

Network info request
Base requests info about the network at startup.

Network info reply
Server tells base network characteristics & RF link settings (no ACK)

6.2. Messages between peers

MPAK to mobile
A base has an MPAK to send to another mobile device. The MPAK may have originated from an FST or another mobile. This includes POSACK MPAKs returned to the mobile. (infoless ACK)

MPAK delivery result
The MPAK was delivered, or could not be delivered due to an error condition. Could be delivered, or failed on account of MAX_REP, inactive, base congestion or other errors. (infoless ACK)

Route obsolete
Sent as a response to *MPAK To mobile* if the mobile is no longer at this base. A-node's route cache entry was obsolete. A-node must contact server for new IP address of where the modem now is and update its route cache. (frame gets no ACK)

7. Overall principle of MPAK routing for mobiles

The objective is to build a cellular wireless network while leveraging existing PC and networking technology as much as possible. As such, using the networking stacks, and IP (TCP/IP and UDP/IP) as much as possible. However, IP does not have any provisions for mobiles roaming to different nodes on a network during the course of a set of interaction.

Objectives

- The network must be able to handle thousands of mobiles being connected to each base at any one time.
- Mobiles can change bases or go in and out of coverage at any time.
- MPAKs must be forwarded to a mobile wherever it is.
- The network must be capable of dealing with the mobile roaming at inopportune times, such as while attempting to deliver an MPAK
- For scalability, payload data (MPAKs) must not be routed through a central node.

There are some things that IP uses to address similar scenarios, but none that does quite what is needed. Existing technologies are:

7.1. Existing similar technologies

Mobile IP

An IP address contains, as part of its address, information about which 'sub-network' the device is currently connected to. This is intentional and necessary as part of the design in order to allow IP routers to not actually know where each individual node (of millions of nodes) is located to route IP packets. As such, in order to connect into a company's IP based LAN (Local Area Network), it is necessary to have an IP address that belongs to that LAN. Using mobile IP, the IP datagrams are tunneled through a TCP connection to the LAN.

However, this technique assumes a semi-permanent IP connection for the duration of a session of usage. As wireless mobiles have a session that is always open, and are able to roam (change their point of connection into the wireless network) even during a course of transaction, Mobile IP is not a usable approach to handle the mobility aspect of a mobile Mobitex device.

DNS (Domain Name Server) resolution system

The DNS system resolves internet domain names (such as those included in a web URL or E-mail address) into physical IP addresses. This is used because it is sometimes necessary for a service to change its IP address. For example, if a server is moved from one location to another physical location, it is often necessary to assign it a new IP address. As such, DNS does allow for some mobility. However, an underlying assumption of DNS is that IP addresses change about as often as phone numbers change (not very often). Again, this does not allow for rapid changes of IP addresses, and most certainly does not allow for connections to remain intact during this process.

7.2. Mobile roaming provisions used by RIM

The objective is to tunnel Mobitex MPAKs through IP. Mobiles are addressed and identified by MAN number (Mobitex Access Number). However, as the packets are delivered through an IP network, they must be routed by IP address.

This necessitates that a base station in the wireless WAN can at any time query which base station a mobile is currently attached to, and what that base station's IP address is. Once the destination base's IP address is established, the MPAK is encapsulated in an IP packet, and sent to that base station.

Because there can be thousands of mobiles on a base station, and that mobiles can roam frequently, the overhead of establishing and keeping TCP/IP connections is not practical. As such, MPAKs are encapsulated instead in connectionless UDP/IP packets. This is possible as the maximum size of an MPAK is smaller than that of a UDP/IP packet. Because UDP is inherently not guaranteed delivery, it is necessary for the receiving base station to send back a UDP packet to indicate that the packet has in fact been received.

There are various scenarios where the base station that the mobile was believed to be on is incorrect, either on account of caching the MAN/IP association, or because the mobile moves to a new base station before the MPAK can be delivered. Such scenarios are unavoidable. This requires that the network be able to deal with routing errors, and subsequently forward it to the correct base station. The following section describes various normal, as well as irregular scenarios in which an MPAK is being delivered.

8. Possible traffic scenarios (these are some twisted scenarios)

Notes:

- Acknowledgements across the RF link are not shown.
- Some acknowledgements across UDP that do not contain new information are also not shown.
- Routing to and from Mobiles is the same as to and from FST (Fixed station terminals). The fact that FSTs are wired to the network and rarely move doesn't make them different.

8.1. Regular MPAK delivery (server not involved)

In this scenario, an MPAK is sent from one mobile to another.

Figure 4 illustrates a regular MPAK delivery. The method comprises of the following steps:

1. A-node receives packet from mobile.
2. A-node verifies that this mobile is allowed to communicate, and is not new from its table of valid mobiles on the base.
3. A-node looks up destination base in route cache.
4. A-node sends *MPAK to mobile* to B-node
5. B-node acknowledges having received the MPAK
6. B-node sends the MPAK to the mobile.

8.2. Mobile sends to mobile route not on base and not in route cache

In this scenario, a mobile sends to another mobile. The base that the mobile is attached to

must look up the location and IP address of the new mobile before sending the MPAK directly to the base station that the other mobile is known to currently be connected to.

Figure 5 illustrates how a mobile sends to a mobile route not on base and not in route cache. The method comprises of the following steps:

1. A-node receives packet from mobile;
2. A-node verifies that this mobile is allowed to communicate, and is not new from its table of valid mobiles on the base;
3. A-node checks route cache and does not find the destination MAN in the list;
4. A-node sends *Mobile Route Request* to server;
5. Server sends *Mobile Route Reply* to base;
6. A-node sends *MPAK to mobile* to B-node;
7. B-node sends ACK to A-node; and
8. B-node sends MPAK to mobile.

8.3. Mobile roams to a new base in absence of traffic:

In this scenario, a mobile moves from one base station to another. The transaction takes place while no MPAKs are pending delivery to a mobile (this is normal, and simplest case of roaming).

Figure 6 illustrates how a mobile roams to a new base in absence of traffic. The method comprises of the following steps:

1. A-node receives first packet from mobile.
2. A-node checks its tables and establishes that this mobile is new.
3. A-node creates new record for this mobile, with subscription data still unknown
4. A-node sends *Mobile Info Request* to server
5. Server sends *Mobile Info Reply* to A-node. If an old location of the mobile is known, the previous base station that the mobile was on is also notified that the mobile has moved on.
6. A-node fills in the record for this mobile

8.4. Mobile sends to mobile that has just moved (route cache obsolete)

In this scenario, a mobile sends to another mobile. The base already has a record for where the destination mobile resides, but this record is no longer correct.

Figure 7 illustrates how a mobile sends to a mobile that has just moved. The method comprises of the following steps:

1. A-node receives packet from mobile.
2. A-node verifies that this mobile is allowed to communicate, and is not new from its table of valid mobiles on the base.
3. A-node checks route cache and determines address of base where it last knew the mobile to reside (in this case, the route cache contains an obsolete value)
4. A-node sends the *MPAK to mobile* to the wrong base (route from cache)
5. B-node that MPAK was sent to looks up destination MAN in its subscription list and sees that it no longer here.
6. Base that MPAK was sent to replies with *route obsolete* to A-node
7. A-node sends *Mobile Route Request* to server

8. Server sends *Mobile Route Reply* to base
9. Base updates route cache and sends *MPAK to mobile* to B-node
10. MPAK is acknowledged
11. B-node sends the MPAK to the mobile.

If mobile has moved again between steps 7 and 10, the process goes back to step 6

8.5. Originating mobile roams while POSAK MPAK is delivered

The method comprises of the following steps:

1. Mobile sends to A-node
2. A-node looks up route to B-node in route cache
3. A-node sends *MPAK to mobile* to B-node
4. B-node confirms that it has received the MPAK to the A-node
5. B-node adds the originating address to its route cache
6. Mobile roams to another base.
7. Other base sends *Mobile Info Request* to server
8. Network sends *Mobile Info Reply* to other base
9. Network sends *Remove Subscription* to A-node
10. B-node delivers the MPAK several seconds later
11. B-node looks up sender in route cache
12. B-node sends *MPAK delivery result* to A-node
13. A-node replies with *route obsolete*
14. B-node sends *Mobile Route Request*
15. Server sends *Mobiele Route Reply* to base
16. B-node sends *MPAK delivery result* to other base
17. Other base acknowledges receipt
18. Other base delivers the POSAK response to originating mobile.

8.6. FST tries to get hold of modem that has just lost coverage

The method comprises of the following steps:

1. Blackberry sends MPAK to FST base
2. FST base looks up route to destination MAN in its route cahce
3. FST base sends *MPAK to mobile* to B-node
4. B-node looks up destination MAN & finds that it is here and should be active
5. B-node attempts to send the MPAK to the mobile several times and gives up
6. B-node base sends *Mpak Delivery Result* to FST base that MPAK delivery has failed.
7. FST base sends *Mobile mail pending* to server
8. Server sends *Mobile info reply* to base where mobile was last known to be, with mailbox info set.
9. Mobile regains coverage, and sees itself listed in SVP5 and sends traffic
10. B-node sends *Mobile active* to server
11. Server informs FST base that mobile is active
12. FST base sends *MPAK to mobile* delivery base
13. B-node delivers MPAK

If MPAK is not POSACK, we are done now. Otherwise, go through MPAK sending procedures to get the MPAK back to the originating subscription.

9. State diagrams of various data items

9.1. A-node MPAK state machine

For purpose of discussion, the A-node is always the base station that the mobile originating an MPAK is attached to. This state machine indicates the states an MPAK may go through before it is moved to the B-node. Once the MPAK is moved to the B-node and acknowledged, its delivery is the B-node's responsibility, and the A-node no longer needs to remember it.

Figure 8 illustrates an A-node MPAK state machine.

Handling the case where both sender and receiver of an MPAK change location while the MPAK is in transit:

A mobile sends an MPAK to another base, and that base acknowledges having received the MPAK. Subsequently, both originating and addressee mobiles roam before the MPAK is delivered. The base that now has the MPAK is neither the sender nor the originator.

This subtle point forces the whole delivery mechanism to be used for any packets that are returned. Otherwise, this case becomes too much of a special case.

B-node doesn't remember that it owns a mobile

When sending a route request, the base must send the IP of the obsolete route it tried to use. If the server sees that the current subscription location of the IP address is on a base is the same as the obsolete route IP address, the base must have forgotten the mobile is there. Subscription will be pushed down to the B-node, and the route sent back to the A-node. There is a possibility that the A-node will get a second obsolete route, but retries should take care of this?

Reasons for failed delivery:

Illegal destination MAN (NO TRANSFER)
Destination MAN out of coverage (NO TRANSFER)
B-node too congested (CONGEST)
Illegal MPAK format (ILLEGAL)
Remote base not responding
Server not responding

9.2. B-node MPAK state machine

This is the state machine for the B-node. Note however that in some circumstances, where the B-node has acknowledged an MPAK and subsequently is unable to deliver it locally, it must forward it onto another base (as indicated by 'stuff it into the originating queue on the diagram). In this case, the B-node now treats the MPAK much like it was the A-node for that MPAK, and uses the A-node state machine for that MPAK.

10. Congestion control

Congestion control is always a hairy area.

Mobitex has some deficiencies in terms of how it handles congestion control. Specifically, an entire FST is throttled back if it only causes a single base in the entire network to become congested. This approach is not acceptable.

IP is much more flexible in terms of congestion control, but IP does not even pretend to be guaranteed delivery.

For our network, there are two types of congestion that will affect performance:

10.1. IP congestion

This type of congestion causes delay or dropping of IP packets. Retries at our UDP link code should handle most of these cases. Also, the protocols are designed end to end where possible, to minimize intermediate hops being stuck with a piece of data that cannot be moved further.
On the whole, IP is assumed to have more bandwidth than is required, so IP congestion should be a relatively rare phenomenon. This is an important assumption, as additional IP bandwidth will be used up when dealing with ROSI congestion

10.2. ROSI congestion

Congestion at the wireless ROSI link level is expected.
Because all access points to the network will be through base stations or FST base stations, the network has the opportunity to throttle back traffic before it hits a congested base. The question is how to do this.
A good approach is for a base with a congested ROSI downlink to return MPAKs indicating that they are congested, and a minimum time before sending to the base should be attempted again. This will be done on a per-MAN number basis. Each MAN number that is trying to send to a base with its queue maxed out will be told to back off before retrying again. If the base were not even able to reply to the UDP packets requesting that the MPAK be sent, dropping the UDP packets would also be legal, as UDP is not expected to be reliable. This would force the UDP link layer to back off as well.

Other problems arise when a mobile is sending MPAKs, and is expecting MPAKs in return. When a mobile has originated an MPAK, its backoff timer should be reduced to zero to allow the base, or remote FST host, to send back something in return to that MPAK.

Problems become especially severe if credit card authorizations, which time out after seconds, are to be run on a congested base. If the base gave temporary priority to man numbers that have recently originated traffic, this case could be handled properly.

11. Needed data fields for messages

11.1. Each UDP datagram shall contain the following fixed fields:

Tag field
Used for matching responses to replies and duplicate elimination. UDP link layer uses this

to match up ACK frames.

Ack flag
This field indicates that this is a response. The UDP link layer uses this to confirm that items were delivered.

Autoack flag
This field indicates that this UDP packet does not generate a response. As such, the UDP link will generate its own ACK. If this flag is not set, the upper layers will generate an immediate response to the packet, which will also serve as the UDP link ACK.

11.2. 32-bit internal MPAK structure fields

All MPAKs will be internally represented by a data structure, refereed to as a 32-bit internal MPAK data type. This data type is optimized for simplified handling of the MPAK internal to the network.

When dealing with existing interfaces, such as the Mobitex ROSI protocol or possibly an X25 link, a translation procedure shall be applied to translate the MPAK into a regular Mobitex MPAK. Once we are ready to expand on the over the air interface, more of the internal MPAK fields will become available across the air.

Originator MAN
This is the 32-bit MAN number that originally generated the MPAK.

Addressee MAN
This is the 32-bit addressee MAN number that is to receive the MPAK

Tag field
A 16 bit field allowing a tag to be attached.

MPAK class / type
8-bit value for Mobitex MPAK class and type. Will add a class for determining the state of another subscription - such as, 'where is...'. Also, 'where is this base'

MPAK flags
8-bit value for Mobitex MPAK flags. Will add flags for truncate posack

MPAK state
8-bit value for Mobitex MPAK state. Will add a state that indicates its a POSAK return MPAK!

Time
This is the 32-bit time value. It indicates the time at which the MPAK was received, in 100'ths of a second since the start of Mobitex time.

HPID
Same as Mobitex HPID...

Payload
Up to 512 bytes of payload data - should we allow bigger downlink MPAKs later?

Time to live
To avoid MPAKs going in circles?

11.3. Subscription state fields

The data fields associated with any subscription on the network. This structure is stored in the server, as well as the base that a mobile is currently on. May also use the exact same structure for the route cache (only the ESN number is redundant)

MAN number
32 bit MAN number field

ESN number info
A 32 bit ESN number. May change the exchanged ESN number to be a hash based on base/area and time of a much larger set of data to prevent subscription thefts?

current Base / Area number
16 bits. Base and area ID of the base that the mobile was last known to be on

current base IP address & port
The IP address of the base that the mobile was last known to be on. This also includes a port number. Specifying a port number will allow us to potentially have several bases sharing the same IP address.

Link Number
For network layers connecting to multiple link layers (if we want to go more tree shaped), this number indicates which of several links the subscription is actually on.

Subscription state
One of several states: Unregistered, Ok, Suspended.

Coverage state
In coverage / out of coverage / location unknown

Mailbox flag & MAN numbers to notify.
Indicates which MAN numbers want to be notified when a mobile comes back in coverage.

Downlink backoff
A value indicating at what time it will be allowed to send traffic to a mobile again. Used for throttling downlink traffic. Not sure how this will work.

12. Encoding the fields in a UDP block

Use Binary representation of UPD packets, as there is no notion of an ASCII session that one could monitor on a terminal anyways. Internal structures all use little endian. Before sending it across IP, the same structure is converted to big endian.

13. Running Linux on embedded system

For a full network, Linux will have to be stripped to run without a hard disk. Instead, some sort of flash file system should be used for this purpose. Ideally, the file system will only have the potential to be in an inconsistent state during firmware upgrades or configuration change.

Another important issue is to keep the system in such a state that it can be completely reset at any time. This will allow some sort of radio-modem based base watching device to reset the base station when it starts to malfunction software wise.

In depth Linux knowledge will be required to make this work.

14. Diagnostics and monitoring facilities

Telnet facilities available. X-app for graphical. HTTP server running on base to get statistics out from anywhere. Statistics as part of defined UDP datagram protocol
Ability to get Rx waveforms from a base. Some sort of bugdisp log or similar from the base code.

It will be appreciated that the above description relates to the preferred embodiment by way of example only. Many variations on the invention will be obvious to those knowledgeable in the field, and such obvious variations are within the scope of the invention as described and claimed, whether or not expressly described.

WHAT IS CLAIMED AS THE INVENTION IS:

1. A base station system comprising:
   a. One or more purpose personal computers wherein each personal computer having:
      i. A sound card configured to the personal computer;
      ii. transmitter and receiver means for RF communication between the personal computer and one or more mobile devices;
      iii. Software modules adapted to the personal computer to operate with the transmitter and receiver means via the sound card.

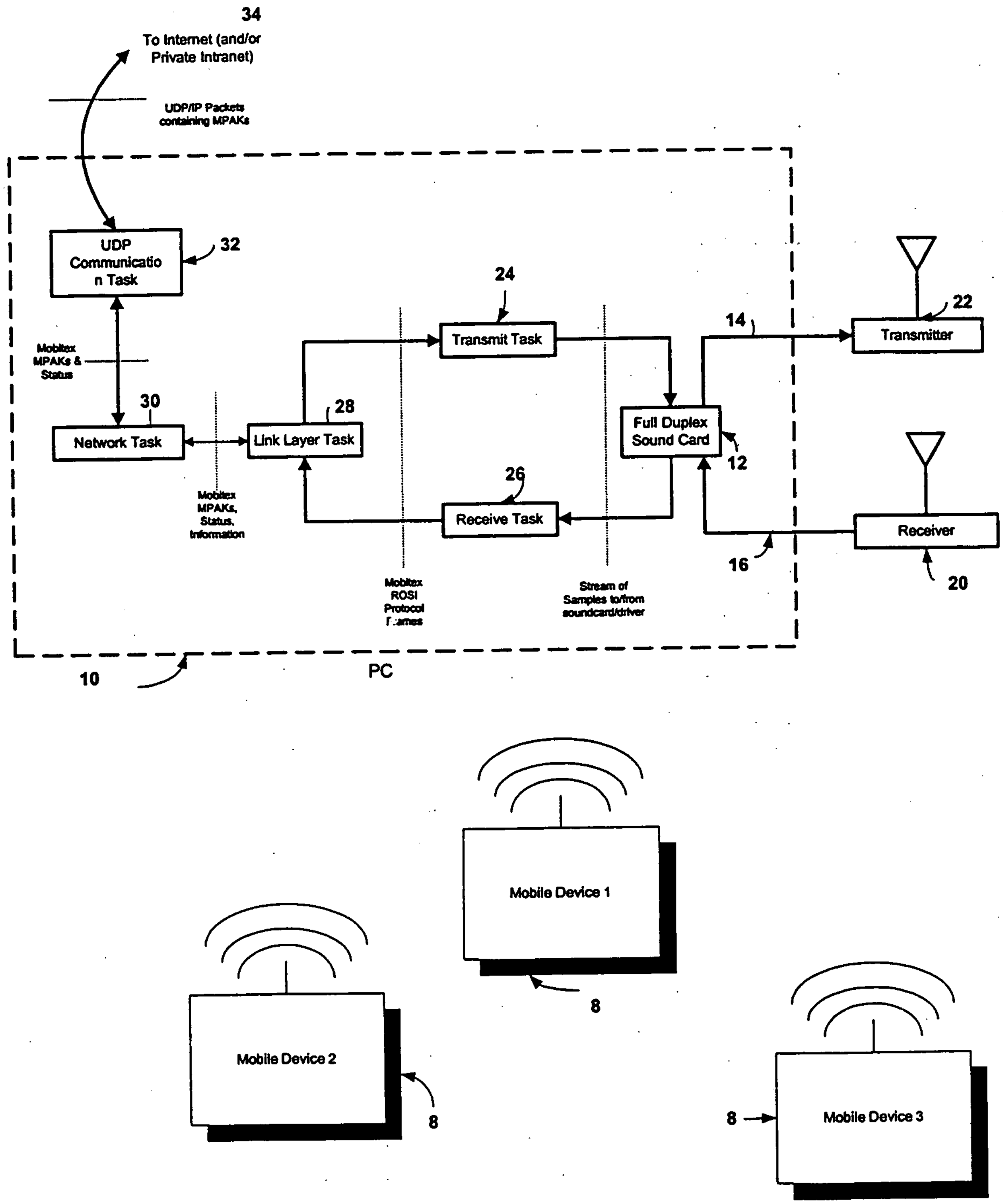
34
To Internet (and/or Private Intranet)
UDP/IP Packets containing MPAKs
UDP Communicatio n Task
32
24
22
Mobitex MPAKs & Status
Transmit Task
14
Transmitter
30
28
Full Duplex Sound Card
Network Task
Link Layer Task
12
26
Mobitex MPAKs, Status, Information
Receive Task
Receiver
16
20
Mobitex ROSI Protocol Frames
Stream of Samples to/from soundcard/driver
10
PC
Mobile Device 1
8
Mobile Device 2
8
8
Mobile Device 3

FIG. 1

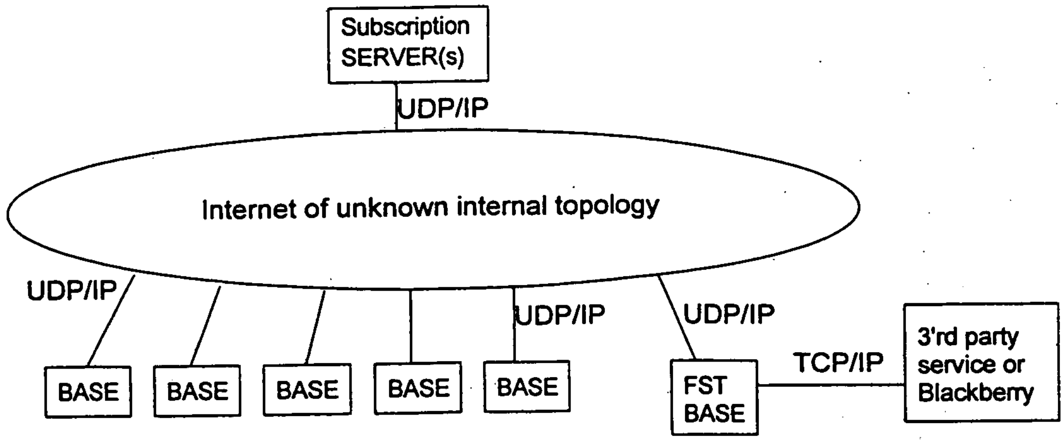

Figure 2

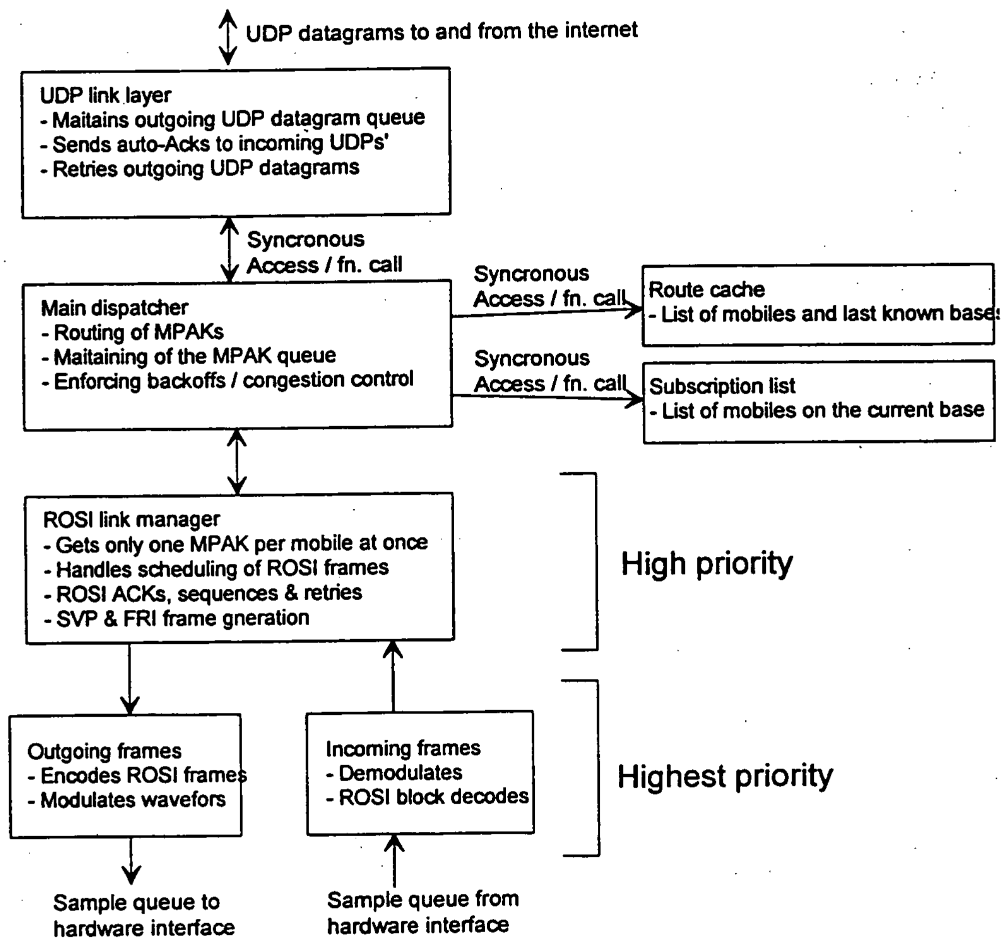

Figure 3

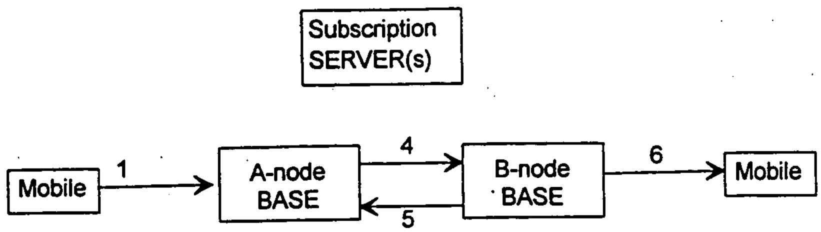

Figure 4

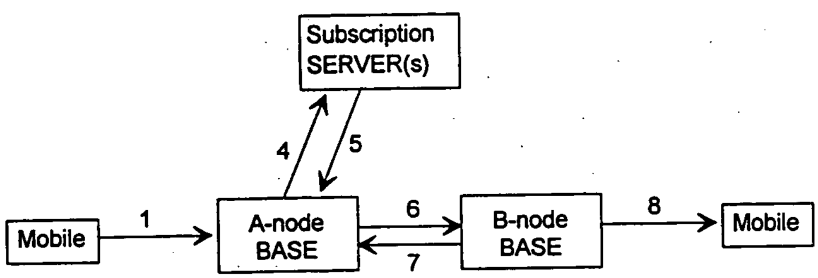

Figure 5

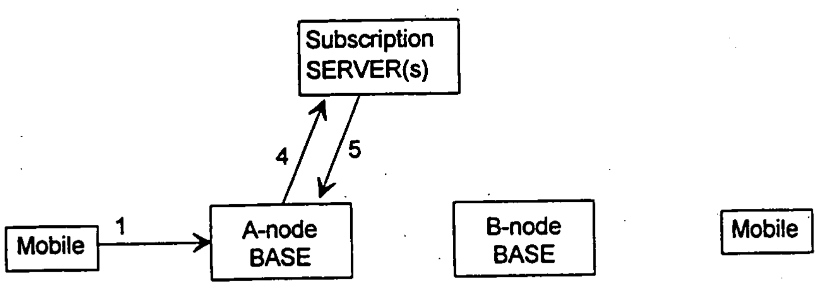

Figure 6

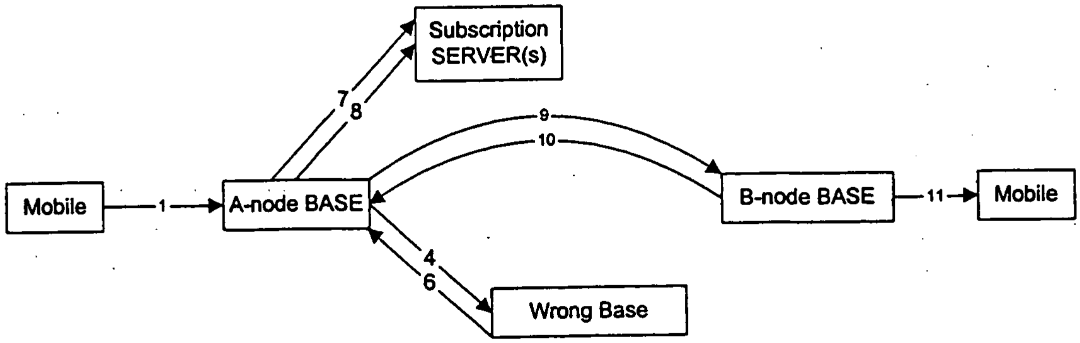

Figure 7

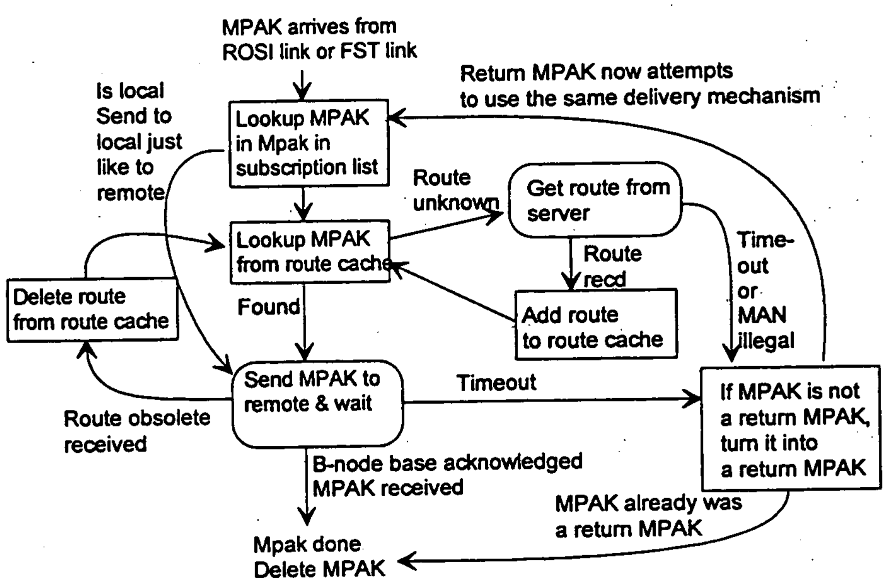

Figure 8

EXHIBIT B

June 20, 2000

Deborah Satoran
Jones, Day, Reavis & Pogue
North Point, 901 Lakeside Avenue
Cleveland, Ohio 44114-1190

Dear Debbie,

Re: New Provisional Patent Application
SYSTEM AND METHOD FOR IMPLEMENTING BASE STATIONS USING PERSONAL COMPUTERS
Your reference: 555255-012-~~150~~ 149

Please find enclosed a clean final draft of the above captioned matter. An electronic copy will be forwarded by email in due course.

If you have any questions, please do not hesitate to contact Krishna Pathiyal at 888-7465 ext. 2535.

Yours truly,

Valerie Robertson

Valerie Robertson
Legal Assistant
Encl.

JDRP
JUN 21 2000
RECEIVED